



NATURAL

Natural

SYSTRANS Utility

Version 5.1.1 for Windows

Version 3.1.6 for Mainframes

Version 5.1.1 for UNIX and OpenVMS

 **SOFTWARE AG**



This document applies to Natural Version 5.1.1 for Windows, Version 3.1.6 for Mainframes, Version 5.1.1 for UNIX and OpenVMS, and to all subsequent releases. Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

© June 2002, Software AG
All rights reserved

Software AG and/or all Software AG products are either trademarks or registered trademarks of Software AG. Other products and company names mentioned herein may be the trademarks of their respective owners.

Table of Contents

SYSTRANS - Introduction	1
SYSTRANS - Introduction	1
General Information	1
Invoking SYSTRANS Online	3
Name and Range Specification	3
SYSTRANS and Entire Connection	4
SYSTRANS and Natural Security	4
Unload Function	5
Unload Function	5
General Unload Options	5
User-Defined Conversion Table	6
Applying the Use Selection List Option	8
Objects to be Unloaded	9
Unloading Natural Objects	9
Unloading Maps	12
Unloading DDMs	13
Unloading Adabas FDTs	14
Unloading Error Message Texts	15
Unloading Command Processors	16
Unloading Libraries	17
Unloading in Batch Mode under OS/390	18
Unloading in Batch Mode under OpenVMS	19
Unloading in Batch Mode under UNIX	20
User Exits for Unloading in Batch Mode	20
Unload Work File Specifications	20
Mainframe	21
OpenVMS, UNIX and Windows	21
Work File for Unload Command Input - Work File 2	21
Fixed Parameter Positions	21
Parameters Separated by a Comma	23
Natural Profile Parameters	23
Load, Scan and Restart Load Functions	25
Load, Scan and Restart Load Functions	25
Load and Scan	25
General Load/Scan Options	25
Objects to be Loaded/Scanned	28
Loading in Batch Mode under OS/390	37
Loading in Batch Mode under OpenVMS	37
Loading in Batch Mode under UNIX	38
User Exits for Loading in Batch Mode	38
Load Work File Specifications	39
Natural Parameter Settings	39
Restart Load	40
Direct Transfer Functions	41
Direct Transfer Functions	41
Direct Transfer using RPC	41
Restart Direct Transfer	43
Get Report from Direct Transfer Load	43
Define Local TRANSFER System	43
Preparing to Use Natural RPC for a Direct Transfer	44
Direct Commands and CALLNAT Interface	45
Direct Commands and CALLNAT Interface	45
Executing SYSTRANS in Direct Command Mode	45

Executing SYSTRANS from within an Application	47
Available Keywords and Functions	49
Functions	49
Objects	49
Parameters	49
Options	50
SYSTRANS Profile	52
SYSTRANS Profile	52
Listing of Text Object PROFILE	52
Parameters in the Text Object PROFILE	54
Unload Parameters	54
Load Parameters	55
Downloading Objects without SYSTRANS	56
Downloading Objects without SYSTRANS	56
Using the Utility FTOUCH	56
Using NFS to Store Natural Libraries	58

SYSTRANS - Introduction

- General Information
 - Invoking SYSTRANS Online
 - Name and Range Specification
 - SYSTRANS and Entire Connection
 - SYSTRANS and Natural Security
-

General Information

The Natural utility SYSTRANS allows you to transfer all Natural objects, maps, DDMs, libraries, command processors and error messages as well as Adabas FDTs from one hardware platform to another.

To be able to do this, the SYSTRANS utility uses a general record layout. This general record layout is independent of all hardware platforms and is used only for transporting objects. Natural objects are read from one hardware platform and then restructured according to the general record layout using the Unload function of the utility.

The formatted records are written to a Natural work file which can be transported to another platform using standard file transfer services. On the target platform, the Natural objects can then be read from the work file, and loaded into the local file or database system using the Load function of SYSTRANS. Natural objects are read from the work file and then restructured according to the structure of the new hardware platform.



Warning:

When you are unloading objects, only one file per unload session is assigned to the work file.

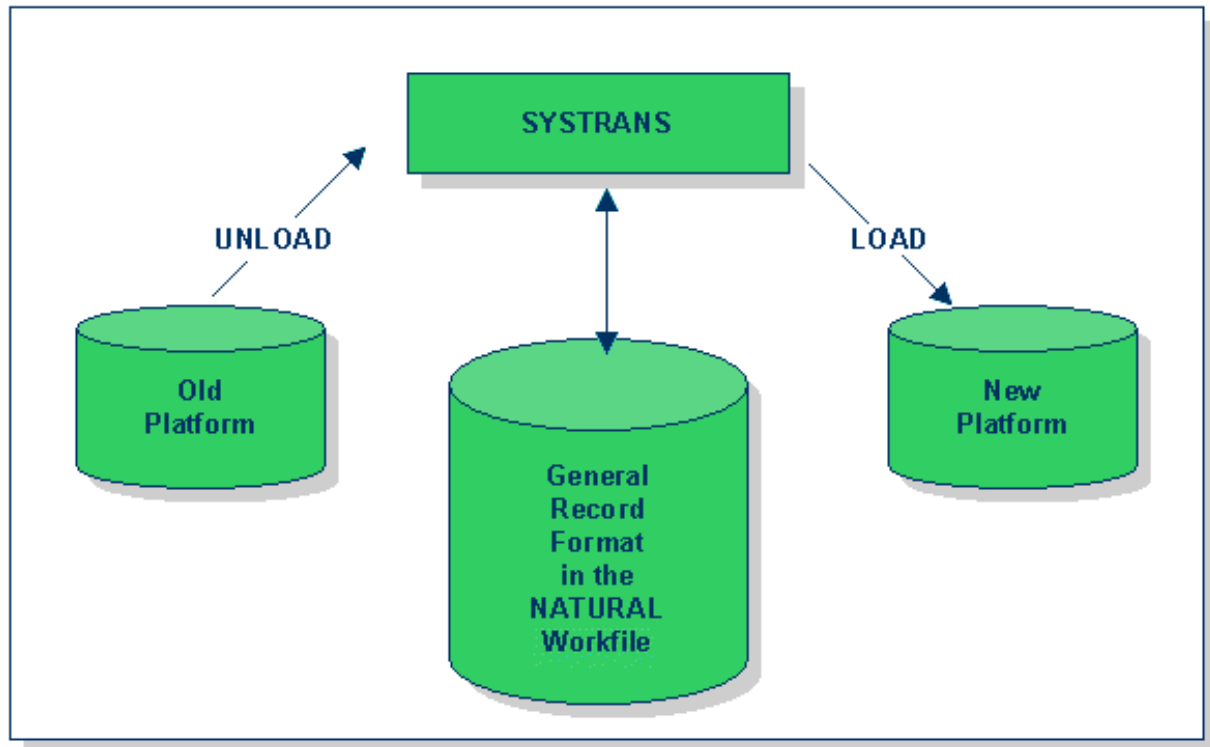
When you are unloading or loading command processors, LFILE 190 must point to an Adabas file for command processor sources.

You can also convert sources from ASCII to EBCDIC or from EBCDIC to ASCII, using either the Unload or Load function (not applicable to mainframes).

Any functionality of the SYSTRANS utility can be restricted by using the user exit TRA-E1-S provided in the library SYSTRANS; for further details, refer to the information provided in the TRA-E1-S source.

You can define a profile for your SYSTRANS utility, both generally and user-specifically. For this purpose, Natural is delivered with the text object PROFILE. To activate this profile, you have to resave PROFILE under the name TRANPROF in the library SYSTRANS. See more information in the section SYSTRANS Profile.

The following diagram shows how the SYSTRANS utility works when transferring a Natural application from one platform to another:



The SYSTRANS functions Unload, Load, Scan and Restart can also be executed in direct-command mode or from within a Natural application; see also the section Direct Commands and CALLNAT Interface.

Invoking SYSTRANS Online

There are two methods for invoking the SYSTRANS utility for interactive usage:

To invoke SYSTRANS online from any Natural library

- Enter the command SYSTRANS.
If you issue a Natural system command from the SYSTRANS command line, the command will apply to the library from which SYSTRANS has been invoked.

To invoke SYSTRANS online from the Natural Main Menu (mainframes only)

1. Select Maintenance and Transfer Utilities.
The corresponding menu is displayed.
2. Select Transfer Objects to Other Platforms.
The SYSTRANS Main Menu is displayed.

The SYSTRANS Main Menu offers the following options:

Unload TRANSFER Objects

Load TRANSFER Objects

Direct TRANSFER Functions

Scan TRANSFER Work File

Restart TRANSFER Load

See the corresponding sections for a description of these functions.

Name and Range Specification

To select Natural objects, maps, DDMs or command processors, you can specify a name or a range of names.

In the list of options below, *value* is any combination of one or more characters:

	Input	Selected Items
Start Value	<i>value</i> >	All items whose names are greater than or equal to <i>value</i> . Example: AB> Selected: AB, AB1, BBB, <u>ZZZZZZZ</u> Not selected: AA1, AAB
End Value	<i>value</i> <	All items whose names are less than or equal to <i>value</i> . Example: AX< Selected: AB, AWW, AX Not selected: AXA, AY
Leading Characters	<i>value</i> *	All items whose names begin with <i>value</i> . Example: AB* Selected: AB, AB1, ABC, ABEZ Not selected: AA1, ACB

SYSTRANS and Entire Connection

You can use Entire Connection for Windows to transfer small applications between sessions in Natural mainframe, OpenVMS or UNIX environments and sessions on the PC. You can also use Entire Connection to unload or load transfer files to or from a PC.

Note:

The size of a library to be transferred by Entire Connection is limited by the available disk space and time. The size of a library can be estimated by adding up the values for "Size in ESIZE" for saved Natural objects, which can be obtained by issuing the command "L DIR *".

If you use Entire Connection, Work File 7 must be defined as Entire Connection work file.

For the definition of the work files, see Unload Work File Specifications (section: Unload Function) and Load Work File Specifications (section: Load, Scan and Restart Load Functions).

Entire Connection automatically converts the transferred data from EBCDIC to ASCII or vice versa; therefore, the SYSTRANS conversion option should not be set to Y.

Note:

Work File 3 does not have to be a PC work file.

Before you start the SYSTRANS utility, you have to enter the command "%+" to enable the Entire Connection communication with the PC.

You invoke and use the Unload Function and the Load, Scan and Restart Load Functions in exactly the same way as described in the corresponding sections.

Once you have specified the parameters for one of the options of either function, you are prompted to enter the name of the file to which the data are to be downloaded.

After completing an unload session, you have to return to the SYSTRANS Main Menu to close Work File 1.

SYSTRANS and Natural Security

For the use of SYSTRANS in a Natural Security environment, see the section Protecting Natural Utilities in the Natural Security documentation. If the old protection mechanism is used, the same restrictions apply as described for SYSMAN in the Natural Security documentation, section System Libraries And Utilities - Old Protection Mechanism.

Unload Function

You use the Unload function to unload source code objects. The Unload function performs the following:

- unloads data for any operating system;
- if requested, converts the data from ASCII to EBCDIC or from EBCDIC to ASCII;
- formats the objects to be unloaded according to the general record layout.

Below is information on:

- General Unload Options
 - Objects to be Unloaded
 - Unloading in Batch Mode under OS/390
 - Unloading in Batch Mode under OpenVMS
 - Unloading in Batch Mode under UNIX
 - User Exits for Unloading in Batch Mode
 - Unload Work File Specifications
 - Natural Profile Parameters
-

General Unload Options

When you invoke the Unload function, you first have to specify the General Unload Options before you can select the objects to be unloaded.

To specify any of the General Unload Options

- In CUI environments:
Enter a Y (Yes) or an N (No).
- In GUI environments:
Select the corresponding option.
Or choose a button.

You can specify the following:

Option	Explanation
Conversion EBCDIC ==> ASCII ASCII ==> EBCDIC	Conversion from EBCDIC to ASCII or from ASCII to EBCDIC is performed, depending on your hardware environment.
User-Defined Conversion Table	Applies if the above Conversion option is selected. If you have specified your own conversion table in the program SULCONV, all data are converted using this conversion table. If the User-Defined Conversion Table option is not selected, all data are converted using a conversion table that is internally defined. See also User-Defined Conversion Table below for more information on defining your own conversion table in the program SULCONV.
Substitute Line References	If line numbers are used as references in the source code, the line numbers of the referenced lines and the line number references are replaced with labels. The sources are not modified in the database.
Report	A report is displayed listing the objects that were unloaded.
Include Line Numbers	By default, line numbers in Natural objects are not transferred. Select this option if you wish to transfer line numbers, too.
Use Work File Input	Input for the Unload function is taken from a work file; see also the section Work File for Unload Command Input (Work File 2).
Use Selection List	If this option is set, a list is displayed where you can select objects, maps, libraries or DDMs to be unloaded; see further details under Applying the Use Selection List Option.
Work File Name	Not applicable to mainframes. See the section Unload Work File Specifications for more information on defining work files. OpenVMS, UNIX: To enter a work file name longer than 35 and up to 253 characters, you must invoke the help function for the Work File Name field.
Use Entire Connection Work File	Not applicable to Windows. All unloaded data are copied to Work File 7 when you exit the Unload function on OpenVMS and UNIX, or directly written to Work File 7 on mainframes. Work File 7 must have been defined as Entire Connection work file in your parameter file.
Browse...	Windows only. A work file selection facility is invoked with a corresponding dialog box.

User-Defined Conversion Table

In the library SYSTRANS, there is the program SULCONV with which you can define your own conversion tables. In this program, you can replace any character to match your hardware environment.

Example:

The following steps are an example of how the ASCII character **A** can be converted to the EBCDIC character **a**:

1. Find out the decimal representation of the ASCII character **A**. In this case, the decimal representation of the ASCII character **A** is 65.
2. Find out the decimal representation of the EBCDIC character **a**. In this case, the decimal representation of the EBCDIC character **a** is 129.
3. Replace the value located at the 65th position of the table BASCE with 129.

Applying the Use Selection List Option

If the Use Selection List unload option is set, a list is displayed which is used to select and unload objects, maps, libraries or DDMs from the specified library.

To set the Use Selection List option

- In CUI environments:
Enter a Y (Yes).
The selection list is displayed after you have made all necessary specifications for the objects to be unloaded as described below.
- In GUI environments:
Select the corresponding option.
Click on the Select... button in the object specification dialog box to invoke the selection list.

To start the unload procedure in CUI environments

1. Either select all items by pressing PF5 (All),
Or select a single item or multiple items by marking it (them) with any character entered in the Cmd column and then press ENTER.
2. Press PF4 (Unld).

To exit, press PF3 or enter a period (.) in the command line.

To start the unload procedure in GUI environments

1. Select all items by choosing the Select All button.
Or select a single item by either double-clicking it,
or clicking it once and choosing the Select button.
Or select multiple items by marking them using the mouse or appropriate key combinations and choosing the Select button.
2. Choose the Unload button.

Choose the Options button if you want to specify any options provided to limit the number of objects displayed in the list; a corresponding dialog box is displayed (see the section Options for the Natural Objects Selection List, Options for the Map Selection List or Options for the DDM Selection List).

Objects to be Unloaded

Once you have specified the General Unload Options, a menu is displayed on which you select one of the object types below to be unloaded. In Windows environments, the Objects to be Unloaded are displayed in the right-hand section of the same menu.

- Natural Objects
- Maps
- DDMs
- Adabas FDTs
- Error Message Texts
- Command Processors
- Library

Unloading Natural Objects

If you select Natural Objects as the type of objects to be unloaded, you can make the following specifications:

S/C Type	For future use; nothing can be specified at the moment.
From Library	The name of the library where the Natural objects are located. If the Use Selection List option is set, you can invoke a selection list of all available libraries and unload them: Press PF1 (CUIs) or open a selection box (GUIs) and select the library.
Object Name	The name of the object to be unloaded See also Name and Range Specification in the section Introduction. The Object Name field does not appear in CUI environments if the Use Selection List option is set. To display a selection list of all objects contained in the specified library: Press ENTER (CUIs) or choose the "Select..." button (GUIs). See also Applying the Use Selection List Option.
Object Type	The type(s) of the object(s) to be unloaded.
To Library	The name of the library into which the objects are to be loaded. If no library is specified, the name specified as From Library is used.

Options for the Natural Objects Selection List

See also the section Applying the Use Selection List Option.

In this selection list, you can limit the number of Natural objects displayed by specifying the following:

CUIs	GUIs	Explanation
Member	Name Range	A single name or a range of names of Natural objects to be listed. See Name and Range Specification in the section Introduction. Additionally, you can use a wildcard (?) within the object name or within the asterisk (*) range.
Types	Types	The types of objects to be listed. To display a selection list of possible object types: CUIs enter a question mark (?) in this field or press PF1 (Help) GUIs choose the "Select..." button
Mode	Programming Mode	The programming mode with which an object has been created. Valid values are: S structured mode; R reporting mode; * both structured and reporting mode or no mode, as with data areas on mainframe computers.
Version		Mainframes only. A mask for the version of objects to be listed. All objects for which the version matches this mask are listed. If an asterisk (*) is specified as version mask, no checks are performed.
User ID	User ID Range	The ID of the user who saved the objects to be listed. If an asterisk (*) is specified as user ID, no checks are performed.

CUIs	GUIs	Explanation										
Save Date	Save Date Range	<p>A date on which or a date range within which an object was saved.</p> <p>Options are:</p> <ul style="list-style-type: none">● a date (1997-06-17)● a starting date range (1997-06>). If you specify a starting date, all Natural objects saved on or after this date are listed;● an ending date range (1997-06<). If you specify an ending date, all Natural objects saved up to this date are listed;● a date mask (1997-06 or 1997-06*). If you specify a date mask, all Natural objects saved during that particular time period are listed. <p>As abbreviations for special dates or date ranges, the following strings can be entered:</p> <table><tr><th>String</th><th>Explanation</th></tr><tr><td><u>T</u>ODAY</td><td>The date of the current day. The day can be followed by +nnnn or -nnnn (where <i>nnnn</i> are numeric digits) and/or by > or < . The resulting date is computed as the date of the current day plus or minus <i>nnnn</i> days.</td></tr><tr><td><u>Y</u>ESTERDAY</td><td>The date of the day before the current day.</td></tr><tr><td><u>M</u>ONTH</td><td>The date range of the current month.</td></tr><tr><td><u>Y</u>EAR</td><td>The date range of the current year.</td></tr></table>	String	Explanation	<u>T</u> ODAY	The date of the current day. The day can be followed by +nnnn or -nnnn (where <i>nnnn</i> are numeric digits) and/or by > or < . The resulting date is computed as the date of the current day plus or minus <i>nnnn</i> days.	<u>Y</u> ESTERDAY	The date of the day before the current day.	<u>M</u> ONTH	The date range of the current month.	<u>Y</u> EAR	The date range of the current year.
String	Explanation											
<u>T</u> ODAY	The date of the current day. The day can be followed by +nnnn or -nnnn (where <i>nnnn</i> are numeric digits) and/or by > or < . The resulting date is computed as the date of the current day plus or minus <i>nnnn</i> days.											
<u>Y</u> ESTERDAY	The date of the day before the current day.											
<u>M</u> ONTH	The date range of the current month.											
<u>Y</u> EAR	The date range of the current year.											
Save Time Range	Save Time Range	<p>A time at which or a time range within which an object was saved.</p> <p>Options are:</p> <ul style="list-style-type: none">● a time (10:11:12)● a starting time range (09>). If you specify a starting time, all Natural objects saved at or after this time are listed;● an ending time range (17:00<). If you specify an ending time, all Natural objects saved up to this time are listed;● a time mask (10: or 10:*). If you specify a time mask, all Natural objects during that particular time period are listed. <p>The time must be specified in the format HH:II:SS (HH = hours, II= minutes, SS = seconds).</p>										

Unloading Maps

If you select Maps as the type of objects to be unloaded, you can make the following specifications:

S/C Type	For future use; nothing can be specified at the moment.
From Library	The name of the library where the maps are located. If the Use Selection List option is set, you can invoke a selection list of all available libraries and unload them: Press PF1 (CUIs) or open a selection box (GUIs) and select the library.
Map Name	The name of the map to be unloaded. See also Name and Range Specification in the section Introduction. The Map Name field does not appear in CUI environments if the Use Selection List option is set. To display a selection list of all maps contained in the specified library: Press ENTER (CUIs) or choose the "Select..." button (GUIs). See also Applying the Use Selection List Option.
Incorporate all Predict Rules	If you set this option and Predict is installed, all Predict rules associated with the map are incorporated into the map source.
Unload associated Free Rules	This option is currently not available.
To Library	The name of the library into which the maps are to be loaded. If no library is specified, the name specified as From Library is used.

Options for the Map Selection List

In this selection list, you can limit the number of maps displayed. Here, the same applies as for the Options for the Natural Objects Selection List with the exception of the Types field which cannot be changed.

Unloading DDMs

If you select DDMs as the type of objects to be unloaded, you can make the following specifications:

S/C Type	<p>GUIs only. For future use; nothing can be specified at the moment.</p>
DDM Name	<p>The name of the DDM to be unloaded. See also Name and Range Specification in the section Introduction.</p> <p>If the Use Selection List option is set, you can invoke a selection list with all DDMs contained in the specified library (OpenVMS, UNIX and Windows) or in the FDIC system file (mainframes): Specify any range as DDM name and press ENTER (CUIs) or choose the "Select..." button (GUIs). See also Applying the Use Selection List Option.</p>
From Library	<p>The name of the library where the DDMs are located. Only to be used in OpenVMS, UNIX and Windows environments. If the Use Selection List option is set, you can invoke a selection list of all available libraries: Press PF1 (OpenVMS, UNIX) or open a selection box (Windows).</p>
To Library	<p>The name of the library into which the DDMs are to be loaded. Only to be used if your target environment is OpenVMS, UNIX or Windows.</p> <p>If no library is specified, the following applies:</p> <ul style="list-style-type: none"> ● If the DDMs are unloaded from a mainframe environment, all DDMs will be automatically loaded into the library SYSTEM. ● If the DDMs are unloaded from any other environment, the name of the library specified as From Library will be used. ● If the target environment is a mainframe environment, any specification is ignored.

Options for the DDM Selection List

See also the section Applying the Use Selection List Option.

In this selection list, you can limit the DDMs displayed by specifying the following items:

Option	Explanation
DDM Name	A single name or a range of names of DDMs to be listed. See Name and Range Specification in the section Introduction.
DDM Range (GUIs)	Additionally, you can use a wildcard (?) within the object name or within the asterisk (*) range.
DBID	The database ID of the DDMs to be listed. If you wish to list only DDMs which have a specific database ID (1), enter that database ID in this field.
FNR	The file number of the DDMs to be listed. If you wish to list only DDMs which have a file number (1), enter that file number in this field. (1) These are the database ID and file number of the physical database file of which the DDM is the logical representation.

Unloading Adabas FDTs

If you select Adabas FDT as the type of objects to be unloaded, you can make the following specifications:

Source DBID	The database ID of the Adabas FDT you want to unload.
Source FNR	The file number of the Adabas FDT you want to unload.
Target DBID	The database ID to be used by the target system for the Adabas FDT. If you enter a 0, the Source DBID specification applies.
Target FNR	The file number to be used by the target system for the Adabas FDT. If you enter a 0, the Source FNR specification applies.
Adabas Password for Source FDT	The appropriate password if your Adabas FDT is password-protected.
Adabas Cipher Code for Source FDT	The appropriate cipher code if your Adabas FDT is protected by a cipher code.

Unloading Error Message Texts

If you select Error Message Texts as the type of objects to be unloaded, you can make the following specifications:

Message Type	<p>The type of error messages to be unloaded:</p> <table> <tr> <th>CUIs</th><th>GUIs</th></tr> <tr> <td>U</td><td>User User-defined error messages</td></tr> <tr> <td>N</td><td>System Natural error messages</td></tr> </table>	CUIs	GUIs	U	User User-defined error messages	N	System Natural error messages
CUIs	GUIs						
U	User User-defined error messages						
N	System Natural error messages						
From Library	<p>The name of the library where the messages are located (only to be used with user-defined error messages). If the Use Selection List option is set, you can invoke a selection list of all available libraries: Press ENTER (CUIs) or open a selection box (GUIs).</p>						
Message Number	The range of error message numbers to be unloaded.						
Language Code	The language code of the error messages to be unloaded. If you specify an asterisk (*), all language codes are unloaded. For valid codes, see the description of the system variable *LANGUAGE in the Natural Programming Reference documentation						
To Library	The name of the library into which the messages are to be loaded (only to be used with user messages). If no library is specified, the name specified as From Library is used.						

Note:

For Natural error messages, you need not specify a library, because they are always unloaded from either the FNAT system file or the error messages subdirectory.

Unloading Command Processors

If you select Command Processors as the type of objects to be unloaded, you can make the following specifications:

S/C Type	For future use; nothing can be specified at the moment.
From Library	The name of the library where the command processors are located. If the Use Selection List option is set, you can invoke a selection list of all available libraries: Press PF1 (CUIs) or open a selection box (GUIs) and select the library.
Command Processor Name	The name of the command processor to be unloaded. See also Name and Range Specification in the section Introduction. The Command Processor Name field does not appear in CUI environments if the Use Selection List option is set. To display a selection list of all command processors contained in the specified library: Press ENTER (CUIs) or choose the "Select..." button (GUIs). See also Applying the Use Selection List Option.
To Library	The name of the library into which the command processors are to be loaded. If no library is specified, the name specified as From Library is used.

Options for the Command Processor Selection List

See also the section Applying the Use Selection List Option.

In this selection list, you can limit the command processors displayed by specifying the following items:

Option	Explanation
Member	A single name or a range of names of command processors to be listed. See Name and Range Specification in the section Introduction.
Command Processor Range (GUIs)	Additionally, you can use a wildcard (?) within the object name or within the asterisk (*) range.

Unloading Libraries

If you select Library as the type of objects of various types to be unloaded from a specific library, you can make the following specifications:

CUIs	GUIs	Explanation
From Library	Library	The name of the library to be unloaded. If the Use Selection List option is set, you can invoke a selection list of all available libraries: Press ENTER (CUIs) or open a selection box (GUIs).
Unload Natural Objects	Natural Objects	Indicates whether the Natural objects contained in the specified library are to be unloaded. Enter a Y for yes (CUIs) or select the corresponding option (GUIs) to unload Natural objects.
Range of Natural Objects	Name Range	The name of the object to be unloaded. See also Name and Range Specification in the section Introduction.
Types of Natural Objects	Types	The type(s) of the object(s) to be unloaded. If you specify an asterisk (*), all types are unloaded.
Unload Maps	Maps	Enter a Y (CUIs) or select the corresponding option (GUIs) to unload maps.
Range of Maps	Name Range	The name of the map to be unloaded. See also Name and Range Specification in the section Introduction.
Unload DDMs	DDMs	Indicates whether the DDMs contained in the specified library are to be unloaded on OpenVMS, UNIX and Windows, or in the FDIC system file on mainframes. Enter a Y (CUIs) or select the corresponding option (GUIs) to unload DDMs.
Range of DDMs	DDM Range	The name of the DDM to be unloaded. See also Name and Range Specification in the section Introduction.
Unload Error Messages	Error Message Texts	Enter a Y (CUIs) or select the corresponding option (GUIs) to unload error messages.
Message Number	Number from . . . to	The range of error message numbers to be unloaded.
Language Code	Language Code	The language code of the error messages to be unloaded. If you specify an asterisk (*), all language codes are unloaded. For valid codes, see the system variable *LANGUAGE in the Natural Programming Reference documentation.
Unload Command Processors	Command Processors	Enter a Y (CUIs) or select the corresponding option (GUIs) to unload command processors.
Range of Command Processors	Range	The name of the command processor to be unloaded. See also Name and Range Specification in the section Introduction.
To Library	To Library	The name of the library into which the objects are to be loaded. If no library is specified, the name specified under From Library is used.

Unloading in Batch Mode under OS/390

The following example shows a JCL procedure which you can use to unload your programs and maps:

```
//*****
//*
//*   UNLOAD Natural SOURCES
//*
//*****
//TRANSFER  JOB CLASS=G,MSGCLASS=X
//NATBAT    EXEC PGM=NATBAT,REGION=3000K,
           PARM='IM=D,INTENS=1'
//STEPLIB   DD DISP=SHR,DSN=NATURAL.LOAD
//          DD DISP=SHR,DSN=ADABAS.LOAD
//CMPRINT   DD SYSOUT=X
//DDCARD    DD *
ADARUN  PROG=USER,DB=10,MODE=MULTI,SVC=249
//CMWKF01   DD DISP=SHR,DSN=WORK.FILE1
//CMWKF03   DD DISP=SHR,DSN=WORK.FILE3
//CMSYNIN   DD *
SYSTRANS
U
N,N,N,Y,N,N,N,N
N
LIB1,*,*,LIB2
M
SRCLIB,*,Y,N,TARLIB
FIN
/*
```

Note:

As shown in the above example, do not specify the S/C Type field when unloading Natural objects, because the S/C Type field is an output field only.

Unloading in Batch Mode under OpenVMS

The following example shows a DCL procedure which you can use to unload your Natural objects (besides maps and DDMs):

```
$ ON ERROR THEN GOTO _error_exit
$ natb := $NATBIN:NATURAL.EXE
$
$ work_dir := mydevice:[mydirectory]
$
$ DEFINE NATOUTPUT SYS$OUTPUT          ! Will be written to the log file
$!
$! Assuming that in the parameter module mytrans, the Work Files 1-4 were
$! assigned to NATWORK01-4
$!
$ DEFINE NATWORK01 'work_dir'natwork01.dat
$ DEFINE NATWORK03 'work_dir'natwork03.dat
$ DEFINE NATWORK04 'work_dir'natwork04.dat
$
$ natb batch parm=mytrans stack=(TRANSCMD U N WHERE NAME * FR LIB1 TO LIB2)
$
$_error_exit:
$ EXIT
```

Unloading in Batch Mode under UNIX

The following example shows a UNIX batch procedure which you can use to unload your Natural objects (besides maps and DDMs), assuming that in the parameter module mytrans, the Work Files 1, 3 and 4 were assigned to the names of your work files:

```
natb batch parm = mytrans stack = "(TRANSCMD U N WHERE NAME * FR LIB1 TO LIB2)"
```

User Exits for Unloading in Batch Mode

Two user exits with which you can handle errors when unloading in batch mode are provided in source form under the names SUL-S-X1 and SUL-S-X2. To be invoked, both must be available as cataloged objects under the names SULEXIT1 and SULEXIT2 in the library SYSTRANS.

SULEXIT1 is invoked if an error occurs that leads to an abnormal termination. It allows you to define a return code.

SULEXIT2 is invoked in the case of error messages or warnings. If it returns a non-zero return code, the unload operation is abnormally terminated; otherwise processing is continued.

Unload Work File Specifications

The following work files are used for unloading:

Work File 1	The file into which the data are unloaded; this work file is always used on OpenVMS, UNIX and Windows. On mainframe computers, this file is only used if the option Use Entire Connection Work File (see General Unload Options) is not specified.
Work File 2	The file which contains the unload commands if you use the use the option Use Work File Input as described under General Unload Options.
Work File 3	The file used for temporary storage while unloading; this work file is only used if the option Use Selection List is set. See also Applying the Use Selection List Option.
Work File 4	The file into which the data are unloaded temporarily when using the Direct Transfer Functions (see the relevant section).
Work File 7	Not applicable to Windows. The file (in Entire Connection format) into which the data are downloaded when specifying the option Use Entire Connection Work File (see General Unload Options).

Mainframe

Use the following JCL parameter values to specify the work files:

Parameter	Work File 1/4	Work File 2	Work File 3
LRECL	96	80	43
RECFM	VB	FB	FB
BLKSIZE	6240	6240	6450

Note:

Since SYSTRANS writes records with a variable length, Work File 1 should be defined with the RECFM value VB to reduce work file size.

OpenVMS, UNIX and Windows

All work files must be of ASCII format. To achieve this, a file extension must be used, but not the file extension ".sag".

Work File for Unload Command Input - Work File 2

Unload commands for command input from Work File 2 can be specified in either of the following ways:

- by specifying the individual parameters at fixed positions,
- by specifying the individual parameters separated by a comma (,).

In both ways, the sequence in which the parameters are specified is the same. The way to be used is determined by the second byte which can be specified as either a blank (for fixed positions) or a comma.

Fixed Parameter Positions

Bytes	Explanation
01 - 01	Object Type: N = Natural object M = Map D = DDM F = Adabas FDT E = Error Message Asterisk notation (*) = comment line
02 - 02	Blank

Bytes	Explanation
03 - 34	<p>Object identification</p> <p>Object = N/M: 03 - 10 Object name or (*); this position is mandatory.</p> <p>Object = D: 03 - 34 Object name or (*); this position is mandatory.</p> <p>Object = E: 03 - 06 Start error message number; this position is mandatory. 07 - 07 Blank 08 - 11 End error message number or blank. 12 - 12 Blank 13 - 20 Language code, blank or (*); blank = default = (*).</p> <p>Object = F: 03 - 07 Source DBID; this position is mandatory. 08 - 08 Blank 09 - 13 Source FNR; this position is mandatory. 14 - 14 Blank 15 - 19 Target DBID or blank; blank = Source DBID. 20 - 20 Blank 21 - 25 Target FNR or blank; blank = Source FNR.</p>
35 - 35	Blank
36 - 43	From Library; this position is mandatory if the object = N/M/D/E (exceptions: D on mainframes and E only if error message type = U or blank).
44 - 44	Blank
45 - 52	To Library or blank, if the object = N/M/D/E; blank = From Library. E only if error message type = U or blank.
53 - 53	Blank

Bytes	Explanation
54 - 70	<p>Additional parameters</p> <p>Object = N: 54 - 68 Object type list, blank or (*); blank = default = (*).</p> <p>Object = M: 54 - 54 Y (Yes), N (No) or blank: Incorporate Predict Rules; blank = default = N. 55 - 55 Blank 56 - 56 Y (Yes), N (No) or blank: Unload associated Free Rules; blank = default = N.</p> <p>Object = F: 54 - 61 Adabas password 62 - 62 Blank 63 - 70 Adabas cipher code</p> <p>Object = E: 54 - 54 N (No), U (User-defined) or blank: error message type; blank = default = U.</p>

Example:

```

N PRGMNAME                                FROMLIBR TOLIBRRY PNSGLAT
M MAPMNAME                                FROMLIBR TOLIBRRY N N
D DDMNAME8901234567890123456789012 FROMLIBR TOLIBRRY
E 1234 5678 12345678                      FROMLIBR TOLIBRRY U
F 12345 12345 12345 12345                  ADAPASSW 12345678

```

Parameters Separated by a Comma

When specifying parameters separated by a comma, the following rules apply:

- The parameters must be separated by a comma (,).
- Blank characters between parameters are not required.
- The sequence of the parameters must be as with fixed positions.
- If a parameter is omitted, a comma (,) must be specified instead.

Example:

```

N,PRGMNAME, FROMLIBR, TOLIBRRY, PNSGLAT
M,MAPMNAME, FROMLIBR, TOLIBRRY, N, N
D,DDMNAME8901234567890123456789012, FROMLIBR, TOLIBRRY
E,1234,5678,12345678, FROMLIBR, TOLIBRRY, U
F,12345,12345,, ADABASPW,12345678

```

Natural Profile Parameters

To be able to use the Unload function, set the following Natural parameters:

Parameter	Mainframe	OpenVMS	UNIX	Windows
MT	0	0	---	---
MADIO	0	0	0	0
MAXCL	0	0	0	0
ESIZE	64	---	---	---

Load, Scan and Restart Load Functions

- Load and Scan
 - Restart Load
-

Load and Scan

Load

You use the Load function to load source code objects. The Load function performs the following:

- loads data from the work file into any system environment;
- if requested, converts the data from ASCII to EBCDIC or from EBCDIC to ASCII;
- gives you the option to load free rules into Predict.

Scan

You use the Scan function to scan for source code objects in Work File 1 without actually loading them.

When you invoke the Load or Scan function, you first have to specify General Options before you can select the objects to be loaded or scanned.

Below is information on:

- General Load/Scan Options
- Objects to be Loaded/Scanned
- Loading in Batch Mode under OS/390
- Loading in Batch Mode under OpenVMS
- Loading in Batch Mode under UNIX
- User Exits for Loading in Batch Mode
- Load Work File Specifications
- Natural Parameter Settings

General Load/Scan Options



To specify any of the following General Load or Scan Options

- In CUI environments:
Enter a Y (Yes) or an N (No).
- In GUI environments:
Select the corresponding option.
Or choose a button.

The following options are provided:

Option	Explanation
User-Defined Conversion Table	<p>If data conversion is required by your target environment: If you set this option and have specified your own conversion table in the program SULCONV, all data are converted using this conversion table. If you do not set this option, all data are converted using a conversion table that is internally defined.</p> <p>Load only: If no data conversion is required by your target environment: CUIs: If you specify F (Force), a table is used that only replaces characters, which means that no actual data conversion takes place. GUIs: See Use ASCII-ASCII Conversion Table below.</p> <p>See User-Defined Conversion Table below for more information on defining your own conversion table in the program SULCONV.</p>
Use ASCII-ASCII Conversion Table	<p>Windows only. Not applicable to the Scan.</p> <p>This option applies if no data conversion is required by your target environment. If you set this option, a table is used that only replaces ASCII characters, which means that no actual data conversion takes place.</p>
Report	<p>This option is not applicable to the Scan. The Scan generates reports automatically.</p> <p>Displays a report listing the objects that were loaded.</p>
Translate Sources to Upper Case	<p>Not applicable to the Scan. Translates any source code to be loaded to upper case.</p>
Save Restart Information	<p>Not applicable to the Scan. Saves restart information while loading. This information can be used to if it has been terminated abnormally; see the section Restart Load Function.</p>
Modify Library Names	<p>Not applicable to the Scan. Enables modification of library name(s) specified as To Library before the actual load operation begins (see also New Library Name below).</p> <p>CUIs: Enter a Y. After you have made all necessary specifications for the items to be loaded, a window is displayed in which you can specify a new library name.</p> <p>GUIs: Select the option and the New Library Name text box appears within each load function dialog box, and you can specify new library names.</p>
Work File Name	<p>Not applicable to mainframes. Specifies a work file name. See Load Work File Specifications for more information on defining work files.</p> <p>Note for OpenVMS and UNIX: To enter a work file name longer than 35 and up to 253 characters, you must invoke the help function for the Work File Name field.</p>
Browse...	<p>Windows only. Invoke a work file selection facility with a corresponding dialog box.</p>

Option	Explanation
Use Entire Connection Work File	<p>Not applicable to Windows. Uploads or scans all data from Work File 7. When using this option, Entire Connection is required, and Work File 7 must have been defined as Entire Connection work file in your parameter file.</p> <p>Note for UNIX and OpenVMS: Since Entire Connection work file usage does not allow any other screen I/Os, the resulting report (see also the Report option above) is written to Work File 4 and will not be displayed until the end of the load or scan operation.</p>

GUIs:

Apart from Report, Translate Sources to Upper Case, and Modify Library Names, all General Load/Scan Options can only be specified once at the beginning of each load session.

New Library Name

If the Modify Library Names option is set, the following applies to the new target library name:

If you have specified a range as library to be loaded, the New Library Name may be a range too, but the number of characters before the asterisk (*) **must not** exceed the number of characters before the asterisk in the Library field.

User-Defined Conversion Table

In the library SYSTRANS, there is the program SULCONV with which you can define your own conversion tables. In this program, you can replace any character to match your hardware environment.

Example:

The following steps are an example of how the EBCDIC character **a** can be converted to the ASCII character **A**:

1. Find out the decimal representation of the EBCDIC character **a**. In this case, the decimal representation of the EBCDIC character **a** is 129.
2. Find out the decimal representation of the ASCII character **A**. In this case, the decimal representation of the ASCII character **A** is 65.
3. Replace the value located at the 129th position of the table BEBCA with 65.

Objects to be Loaded/Scanned

Once you have specified the General Load/Scan Options, a menu is displayed on which you select one of the below object types to be loaded or scanned. In Windows environments, the Objects to be Loaded/Scanned are displayed in the right-hand section of the same menu.

- Natural Objects
- Maps
- DDMs
- Adabas FDTs
- Error Message Texts
- Command Processors
- Library
- All Objects

Loading/Scanning Natural Objects

If you select Natural Objects as the type of objects to be loaded or scanned, you can make the following specifications:

S/C Type	For future use; nothing can be specified at the moment.
Library	<p>The library to be loaded or scanned. If you specify no library, all libraries are loaded or scanned. You can specify a group of libraries by using asterisk notation (*). You can only specify a library that has been specified as target library with the Unload function. If no target library has been specified, the name of the library specified as From Library is used.</p> <p>If the Modify Library Names option (see General Load/Scan Options) is set, you can enter a new library name before the actual load operation begins (GUIs: see also New Library Name below).</p>
Object Name	<p>The name of the object to be loaded or scanned.</p> <p>See also Name and Range Specification in the section Introduction.</p>
Object Type	The type(s) of object(s) to be loaded or scanned.
Replace	<p>Not applicable to the Scan.</p> <p>If you specify Y (GUIs: Yes) and a source with the same name as the one you are loading already exists in the target environment, the target source is replaced.</p>
New Library Name	<p>GUIs only.</p> <p>Not applicable to the Scan.</p> <p>The name of the new target library. If the Modify Library Names option (see General Load/Scan Options) is set, the name of the specified target library can be modified here; see also Library above. If not, the New Library Name text box is not available.</p>

Loading/Scanning Maps

If you select Maps as the type of objects to be loaded or scanned, you can make the following specifications:

S/C Type	For future use; nothing can be specified at the moment.
Library	<p>The library to be loaded or scanned. If you specify no library, all libraries are loaded or scanned. You can specify a group of libraries by using asterisk notation (*). You can only specify a library that has been specified as target library with the Unload function. If no target library has been specified, the name of the library specified as From Library is used.</p> <p>If the Modify Library Names option (see General Load/Scan Options) is set, you can enter a new library name before the actual load operation begins (GUIs: see also New Library Name below).</p>
Map Name	<p>The name of the map to be loaded or scanned for.</p> <p>See also Name and Range Specification in the section Introduction.</p>
Load Free Rules to Predict	<p>Not applicable to the Scan.</p> <p>If you specify Y (Yes), all free rules on the work file are loaded to Predict.</p>
Replace	<p>Not applicable to the Scan.</p> <p>If you specify Y (GUIs: Yes) and a map with the same name as the one you are loading already exists in the target environment, the target map is replaced.</p>
New Library Name	<p>GUIs only.</p> <p>Not applicable to the Scan.</p> <p>The name of the new target library. If the Modify Library Names option (see General Load/Scan Options) is set, the name of the specified target library can be modified here; see also Library above. If not, the New Library Name text box is not available.</p>

Loading/Scanning DDMs

If you select DDMs as the type of objects to be loaded or scanned, you can make the following specifications:

DDM Name	The name of the DDM to be loaded or scanned for. See also Name and Range Specification in the section Introduction.
Library	<p>The name of the library to be loaded or scanned. Only to be used if your target environment is OpenVMS, UNIX or Windows.</p> <p>You can only specify a library that has been specified as target library with the Unload function. If no library is specified, the following applies to the load procedure:</p> <ul style="list-style-type: none"> ● If the DDMs are unloaded from a mainframe environment, all DDMs will be automatically loaded into the library SYSTEM. ● If the DDMs are unloaded from any other environment, the name of the library specified as From Library will be used. <p>Scan: If you specify no library, all DDMs are scanned.</p> <p>If the Modify Library Names option (see General Load/Scan Options) is set, you can enter a new library name before the actual load operation begins (GUIs: see also New Library Name below).</p>
Replace	<p>Not applicable to the Scan.</p> <p>If you specify Y (GUIs: Yes) and a DDM with the same name as the one you are loading already exists in the target environment, the target DDM is replaced.</p>
New Library Name	<p>GUIs only. Not applicable to the Scan.</p> <p>The name of the new target library. If the Modify Library Names option (see General Load/Scan Options) is set, the name of the specified target library can be modified here; see also Library above. If not, the New Library Name text box is not available.</p>

Loading/Scanning Adabas FDTs

If you select Adabas FDTs as the type of objects to be loaded or scanned, you can make the following specifications:

DBID	The database ID of the Adabas FDT you want to load or scan.
FNR	The file number of the Adabas FDT you want to load or scan.
Replace	Not applicable to the Scan. If you specify Y (GUIs: Yes) and an FDT with the same DBID and FNR as the one you are loading already exists in the target environment, the target FDT is replaced.

If DBID and FNR are left blank or set to **0**, all FDTs on the work file are loaded.

Note for UNIX, OpenVMS and Windows:

All FDTs are written to Work File 5, which you then can use as input for the Adabas utility ADAFDU.

Loading/Scanning Error Message Texts

If you select Error Message Texts as the type of objects to be loaded or scanned, you can make the following specifications:

Message Type	<p>The type of error messages to be loaded or scanned for:</p> <table><tr><th>CUIs</th><th>GUIs</th><th></th></tr><tr><td>U</td><td>User</td><td>User-defined error messages</td></tr><tr><td>N</td><td>System</td><td>Natural error messages</td></tr><tr><td>*</td><td>All</td><td>All error messages</td></tr></table>	CUIs	GUIs		U	User	User-defined error messages	N	System	Natural error messages	*	All	All error messages
CUIs	GUIs												
U	User	User-defined error messages											
N	System	Natural error messages											
*	All	All error messages											
Library	<p>The library to be loaded or scanned. If you specify no library, all libraries are loaded or scanned. You can specify a group of libraries by using asterisk notation (*). You can only specify a library that has been specified as target library with the Unload function. If no target library has been specified, the name of the library specified as From Library is used.</p> <p>If the Modify Library Names option (see General Load/Scan Options) is set, you can enter a new library name before the actual load operation begins (GUIs: see also New Library Name below).</p>												
Message Number	The range of error message numbers to be loaded or scanned.												
Language Code	The language code of the error messages to be loaded or scanned; for valid codes, see the *LANGUAGE system variable in the Natural Programming Reference documentation.												
Replace	<p>Not applicable to the Scan.</p> <p>If you specify Y (GUIs: Yes) and a message with the same number as the one you are loading already exists in the target environment, the target message is replaced.</p>												
New Library Name	<p>GUIs only.</p> <p>Not applicable to the Scan.</p> <p>The name of the new target library. If the Modify Library Names option (see General Load/Scan Options) is set, the name of the specified target library can be modified here; see also Library above. If not, the New Library Name text box is not available.</p>												

Note:

For Natural error messages, you need not specify a library, because they are always loaded into the FNAT system file or the error messages subdirectory.

Loading Command Processors

If you select Command Processors as the type of objects to be loaded or scanned, you can make the following specifications:

Library	<p>The library to be loaded or scanned. If you specify no library, all libraries are loaded or scanned. You can specify a group of libraries by using asterisk notation (*). You can only specify a library that has been specified as target library with the Unload function. If no target library has been specified, the name of the library specified as From Library is used.</p> <p>If the Modify Library Names option (see General Load/Scan Options) is set, you can enter a new library name before the actual load operation begins (GUIs: see also New Library Name below).</p>
Object Name (GUIs: NCP Name)	<p>The name of the command processor to be loaded or scanned.</p> <p>See also Name and Range Specification in the section Introduction.</p>
Replace	<p>Not applicable to the Scan.</p> <p>If you specify Y (GUIs: Yes) and a source with the same name as the one you are loading already exists in the target environment, the target source is replaced.</p>
New Library Name	<p>GUIs only.</p> <p>Not applicable to the Scan.</p> <p>The name of the new target library. If the Modify Library Names option (see General Load/Scan Options) is set, the name of the specified target library can be modified here; see also Library above. If not, the New Library Name text box is not available.</p>

Loading/Scanning Libraries

If you select Library as the type of objects to be loaded or scanned, you can make the following specifications:

CUIs	GUIs	Explanation
Library	Library	<p>The library to be loaded or scanned. If you specify no library, all libraries are loaded or scanned. You can specify a group of libraries by using asterisk notation (*). You can only specify a library that has been specified as target library with the Unload function. If no target library has been specified, the name of the library specified as From Library is used.</p> <p>If the Modify Library Names option (see General Load/Scan Options) is set, you can enter a new library name before the actual load operation begins (GUIs: see also New Library Name below).</p>
Load/Scan Natural Objects	Natural Objects	<p>Indicates whether the Natural objects unloaded for the specified library are to be loaded or scanned.</p> <p>Enter a Y for yes (CUIs) or select the corresponding option (GUIs) to load Natural objects.</p>
Range of Natural Objects	Name Range	<p>The name of the object to be loaded or scanned. See also Name and Range Specification in the section Introduction.</p>
Types of Natural Objects	Types	<p>The type(s) of the object(s) to be loaded or scanned.</p>
Load/Scan Maps	Maps	<p>Indicates whether the maps unloaded for the specified library are to be loaded or scanned.</p> <p>Enter a Y (CUIs) or select the corresponding option (GUIs) to load maps.</p>
Range of Maps	Name Range	<p>The name of the map to be loaded or scanned. See also Name and Range Specification in the section Introduction.</p>
Load/Scan DDMs	DDMs	<p>Indicates whether the DDMs unloaded for the specified library are to be loaded or scanned on OpenVMS, UNIX and Windows. On mainframes, all DDMs on the work file are affected.</p> <p>Enter a Y (CUIs) or select the corresponding option (GUIs) to load DDMs.</p>
Range of DDMs	DDM Range	<p>The name of the DDM to be loaded or scanned. See also Name and Range Specification in the section Introduction.</p>
Load/Scan Error Messages	Error Message Texts	<p>Indicates whether the error message texts unloaded for the specified library are to be loaded or scanned.</p> <p>Enter a Y (CUIs) or select the corresponding option (GUIs) to load or scan error messages.</p>
Message Number	Number from . . . to	<p>The range of error message numbers to be loaded or scanned.</p>

CUIs	GUIs	Explanation
Language Code	Language Code	The language code of the error messages to be loaded or scanned. For valid codes, see the system variable *LANGUAGE in the Natural Programming Reference documentation.
Load/Scan Command Processors	Command Processors	Indicates whether the Command Processors unloaded for the specified library are to be loaded or scanned. Enter a Y (CUIs) or select the corresponding option (GUIs) to load or scan for DDMs.
Range of Command Processors	Name Range	The name of the command processor to be loaded or scanned. See also Name and Range Specification in the section Introduction.
Replace	Replace	Not applicable to the Scan. If you specify Y (GUIs: Yes) and a source with the same name as the one you are loading already exists in the target environment, the target source is replaced.
	New Library Name	GUIs only. Not applicable to the Scan. The name of the new target library. If the Modify Library Names option (see General Load/Scan Options) is set, the name of the specified target library can be modified here; see also Library above. If not, the New Library Name text box is not available.

Loading/Scanning All Objects

Scan

If you select All Objects (that is, if you want to scan for all objects on the work file), no further parameters need to be specified.

Load

If you select All Objects for loading (that is, if you want to load all objects contained on the work file), you can specify the following:

Replace	If you specify Y (GUIs: Yes) and an object with the same name as the one you are loading already exists in the target environment, the target object is replaced.
New Library Name	GUIs only. The name of the new target library. If the Modify Library Names (see General Load/Scan Options) is set, the name of the specified target library can be modified here. If not, the New Library Name text box is not available.

CUIs:

All objects on the work file are loaded into the target libraries as specified with the Unload function, unless you have set the Modify Library Names (see General Load/Scan Options) option. If so, a second window is displayed, in which you can enter a new library name.

Note for OpenVMS, UNIX and Windows:

For Adabas FDTs, the Load All Objects function uses an additional work file.

For more information on defining work files, see the section Load Work File Specifications.

Loading in Batch Mode under OS/390

The following example shows a JCL procedure which you can use to load your programs and maps:

```
//*****
//*
//*      LOAD NATURAL SOURCES
//*
//*****
//TRANSFER JOB CLASS=G,MSGCLASS=X
//NATBAT EXEC PGM=NATBAT,REGION=3000K,
//          PARM='IM=D,MADIO=0,MT=0'
//STEPLIB DD DISP=SHR,DSN=NATURAL.LOAD
//          DD DISP=SHR,DSN=ADABAS.LOAD
//CMPRINT DD SYSOUT=X
//DDCARD DD *
ADARUN PROG=USER, DB=010,MODE=MULTI,SVC=249
//CMWKF01 DD DISP=SHR,DSN=WORK.FILE1
//CMSYNIN DD *
SYSTRANS
L
N,Y,N,N,Y,N
*
N
NEWLIBS
FIN
/*
```

Note:

As shown in the above example, do not specify the S/C Type field when loading Natural objects, because the S/C Type field is an output field only.

Loading in Batch Mode under OpenVMS

The following example shows a DCL procedure which you can use to load your programs and maps:

```
$ ON ERROR THEN GOTO _error_exit
$ natb := $NATBIN:NATURAL.EXE
$
$ work_dir := mydevice:[mydirectory]
$
$ DEFINE NATOUTPUT SYS$OUTPUT ! Will be written to the log file
$!
$! Assuming that in the parameter module mytrans, the Work Files 1-4 were
$! assigned to NATWORK01-4
$!
$ DEFINE NATWORK01 'work_dir'natwork01.dat
$ DEFINE NATWORK03 'work_dir'natwork03.dat
$ DEFINE NATWORK04 'work_dir'natwork04.dat
$
$ natb batch parm=mytrans stack=(TRANSCMD LOAD ALL WHERE NEW-LIBRARY NEWLIBS)
$
$_error_exit:
$ EXIT
```

Loading in Batch Mode under UNIX

The following example shows a UNIX batch procedure which you can use to load your Natural objects (besides maps and DDMs), assuming that in the parameter module mytrans, the Work Files 1, 3 and 4 were assigned to the names of your work files:

```
natb batch parm = mytrans stack = "(TRANSCMD LOAD ALL WHERE NEW-LIBRARY NEWLIBS)"
```

User Exits for Loading in Batch Mode

Two user exits with which you can handle errors when loading in batch mode are provided in source form under the names SLD-S-X1 and SLD-S-X2. To be invoked, both must be available as cataloged objects under the names SLDEXIT1 and SLDEXIT2 in the library SYSTRANS.

SLDEXIT1 is invoked if an error occurs that leads to an abnormal termination. It allows you to define a return code.

SLDEXIT2 is invoked in the case of error messages or warnings. If it returns a non-zero return code, the load operation is abnormally terminated; otherwise processing is continued.

Load Work File Specifications

The following work files are used for loading:

Mainframes

Work File 1	The file from which the data are loaded.
Work File 4	The file into which the load report is written when using the Direct Transfer Functions.

Use the following JCL parameter values to specify the work file:

Parameter	Value
LRECL	96
RECFM	VB
BLKSIZE	6240

OpenVMS, UNIX and Windows

Work File 1	The file from which the data are loaded; it must have a record length of 96. If this work file is transferred via tape, use the UNIX command dd to get the record correctly blocked.
Work File 4	The file into which the load report is written when using the Direct Transfer Functions as described below.
Work File 5	The file which is used to load the FDTs for further processing.
Work File 6	The file which is used to save the restart information.
Work File 7	Not applicable to Windows. The file (in Entire Connection format) from which the data are uploaded when specifying the option "Use Entire Connection Work File".

All work files must be of ASCII format. To achieve this, a file extension must be used, but not the file extension ".sag".

Natural Parameter Settings

To be able to use the Load function, set the following Natural parameters:

Parameter	Mainframe	OpenVMS	UNIX	Windows
MT	0	0	---	---
MADIO	0	0	0	0
MAXCL	0	0	0	0

Restart Load

You use the Restart Transfer Load function to resume a load operation that has been terminated abnormally.

To be able to use this function, the load option Save Restart Information must have been set (see General Load/Scan Options). If so, the necessary restart information (that is, the work file header and all relevant load parameters) is saved either in a Natural text object called LOAD-LOG (on mainframes) or on work file 6 (in all other environments).

When you invoke the Restart Transfer Load function, you first have to specify the following general restart options:

Work File Name	Here you can specify a work file name. Note: This option does not appear in a mainframe environment. To enter a work file name longer than 35 and up to 253 characters, you must invoke the help function for the Work File Name field.
Browse...	Windows only. If you choose this button, a work file selection facility is invoked with a corresponding dialog box.
Use Entire Connection Work File	Not applicable to Windows. Enter Y (Yes) if the data to be loaded are to be uploaded from work file 7. When using this option, Entire Connection is required, and Work File 7 must have been defined as Entire Connection work file in your parameter file.

The Restart Load function checks whether the work file header stored as part of the restart information corresponds to that of the work file to be (re)loaded.

- If so, the Restart Load function resumes the previously terminated load operation.
- If not, the Restart Load function cannot be performed and a corresponding message is displayed. The same applies if you select the Restart Load function without the Save Restart Information option set to Y.

Direct Transfer Functions

With the Direct Transfer Functions, you can unload and load source-code objects by using the Natural RPC (Remote Procedure Call) facility and the SYSRPC Utility as described in the relevant documentation.

When you invoke the Direct Transfer Functions, you can select the following:

- Direct Transfer using RPC
- Restart Direct Transfer
- Get Report from Direct Transfer Load
- Define Local TRANSFER System
- Preparing to Use Natural RPC for a Direct Transfer

Direct Transfer using RPC

A direct transfer starts a remote server, which unloads local data to a remote work file (Work File 1) and then loads the data into the corresponding remote environment.

Before you select the objects to be transferred on the Unload Objects menu, you have to specify the following direct transfer options for defining your transfer session with Natural RPC:

Direction	CUIs only. An output field that specifies the transfer direction (for future use).
Remote TRANSFER System Number	A four-digit code (<i>nnnn</i>) identifying the location of the remote subprogram TRPC <i>nnnn</i> , which is to perform the transfer service. Note: So that you can perform a remote transfer, a client stub of the same name as the subprogram TRPC <i>nnnn</i> must be available; see the section Preparing to Use Natural RPC for a Direct Transfer.
Unload Options	For the individual unload options, refer to the General Unload Options in the section Unload Function.
Load Options	For the individual load options, refer to the General Load/Scan Options in the section Load, Scan and Restart Load Functions. Note: As Software AG'S Entire Broker is involved when using Natural RPC, data conversion is automatically done by the Entire Broker and not by SYSTRANS. Therefore, selecting the option "Load with User Conversion Table" means that in addition to the Entire Broker data conversion, a table is used to replace certain characters by others. This table (SULAS-AS or SULEB-EB) is part of the program SULCONV located in your remote environment.
Replace	If you specify Y (GUIs: Yes) and an object with the same name as the one you are loading already exists in the remote target environment, the target object is replaced.

GUIs:

Once you have specified the direct transfer options, you can select one of the following object types to be unloaded for the direct transfer:

- Natural Objects
- Maps
- DDMs
- Adabas FDT
- Error Message Texts
- Command Processors
- Library

After each unload operation, you are returned to the Direct Transfer Function dialog box, which now shows a Load button (instead of the Exit button) that you must choose to explicitly start the load operation. You can perform multiple unload operations before you start the load operation.

CUIs:

Once you have specified the direct transfer options, the Unload Objects menu (see Objects to be Unloaded in the section Unload Function) is displayed, where you can select the objects to be unloaded.

As the load operation starts automatically after the unload operation is finished, the Load function is not explicitly invoked and you therefore have to specify the individual load parameters before you start unloading your objects. Therefore, when you return to the Unload Object menu, Load (instead of Exit) is assigned to PF3, which means that pressing PF3 both ends the unload session and starts the load operation.

Restart Direct Transfer

You use the Restart Direct Transfer function to resume a direct transfer that has been terminated abnormally. When you select this function, you first have to specify your Remote TRANSFER System Number (see Direct Transfers using RPC).

To be able to restart a remote transfer:

- A client stub of the same name as the subprogram `TRPCnnnn` must be available; see Preparing to Use Natural RPC for a Direct Transfer.
- The direct transfer load option Save Restart Information (see General Load/Scan Options) must have been selected. Otherwise, the direct transfer cannot be resumed and a window (GUIs: dialog box) is displayed showing you a corresponding message and asking you whether to perform the entire load operation again. If so, the Direct Transfer Reload window (GUIs: dialog box) is displayed, in which you can again specify the relevant load options as well as the Replace option.

See the Restart Load function for further information on restarting a data transfer.

Get Report from Direct Transfer Load

You use the function "Get Report from Direct Transfer Load" to obtain the load report, which is written to Work File 4 in the remote environment.

If you select this function, a window (GUIs: dialog box) is displayed in which you enter your Remote TRANSFER System Number (see Direct Transfers (using RPC)). Then, press ENTER (GUIs: choose OK) to have the contents of Work File 4 displayed on a screen (GUIs: list box). It displays a listing of the objects that have been loaded.

Define Local TRANSFER System

This function is only available in environments which support the Natural RPC servers.

You use the Define Local TRANSFER System function to specify your own TRANSFER system number with which you can be addressed as a server by other clients. You can define several servers (that is, TRANSFER systems), but only one system number per server.

When you select this function, you have to specify a four-digit Local TRANSFER System Number (*mmmm*) along with the DBID (database identification) and FNR (file number) of the target system file. The subprogram `TRPCmmmm` is then copied into the library SYSTEM on the specified system file.

Preparing to Use Natural RPC for a Direct Transfer

To use Natural RPC to perform a direct transfer

1. Define your TRPC_{nnnn} subprogram in the service directory.
2. Generate the corresponding client stub and copy the generated stub in the library SYSTRANS.
The client stub must have the same name as the TRPC_{nnnn} subprogram; the parameters to be passed are provided in the parameter data area TRPCPDA, which is delivered in source form.
For further information see Creating Stub Subprograms (Natural RPC documentation) and the SYSRPC Utility documentation.
3. Start the corresponding server.

Direct Commands and CALLNAT Interface

- Executing SYSTRANS in Direct Command Mode
 - Executing SYSTRANS from within an Application
 - Available Keywords and Functions
-

Executing SYSTRANS in Direct Command Mode

In direct command mode, SYSTRANS functions are invoked by the command TRANSCMD (provided in the library SYSTEM) using the following syntax:

```
TRANSCMD function object parameters options
```

Multiple direct command strings can be specified with TRANSCMD. The individual parts of a SYSTRANS command string are defined by keywords. The corresponding values can be entered in upper or lower case. The sequence of parameters and options is variable, since individual values are identified by keywords, and not all keywords must be specified (see Available Keywords and Functions).

Example 1:

If TRANSCMD is issued from the Natural command line (or via the Natural stack), and all keywords and corresponding values directly follow the command name in the same input line, the keywords and values must be separated either by blanks or by the input delimiter (defined by the Natural profile parameter ID).

Example for issuing TRANSCMD from the Natural command line or via the Natural stack (assuming ID=','):

```
TRANSCMD LOAD NAT-OBJECT LIBRARY mylib NAME pgm*
```

or:

```
TRANSCMD LOAD,NAT-OBJECT,LIBRARY,mylib,NAME,pgm*
```

Example for issuing TRANSCMD via the Natural stack from within a Natural program (assuming ID=','):

```
STACK TOP COMMAND 'TRANSCMD LOAD,NAT-OBJECT,LIBRARY,mylib,NAME,pgm*'
END
```

Example 2:

If TRANSCMD is issued in batch mode (or via the Natural stack), and keywords and corresponding values do not directly follow the command name in the same line, the keywords and values must be separated by the input delimiter character (defined by the Natural profile parameter ID).

Example for issuing TRANSCMD via the Natural stack from within a Natural program (assuming ID=', '):

```
STACK TOP DATA 'LOAD,DDM,LIBRARY,mylib,NAME,dm*'
STACK TOP DATA 'LOAD,NAT-OBJECT,LIBRARY,mylib,NAME,pgm*'
STACK TOP COMMAND 'TRANSCMD'
END
```

Example for issuing TRANSCMD in mainframe batch mode (assuming ID=', '):

```
TRANSCMD
LOAD,NAT-OBJECT,LIBRARY,mylib,NAME,pgm*
LOAD,DDM,LIBRARY,mylib,NAME,dm*
```

Executing SYSTRANS from within an Application

To invoke functions of SYSTRANS from within a Natural application, the Natural subprogram TRANSIF is provided in the library SYSTEM. Since only one SYSTRANS function can be executed at a time, subsequent calls to TRANSIF are necessary if several commands are to be processed, for example, when unloading.

The required SYSTRANS command syntax is as follows:

function object parameters options

The individual parts of a SYSTRANS command are defined by keywords. The keywords and corresponding values must be separated by blanks. Values can be entered in upper or lower case. The sequence of parameters (see Parameters below) and options (see Options below) is variable, since individual values are identified by keywords, and not all keywords must be specified.

The parameter data used in TRANSIF are described in the following table:

Parameter	Format/Length	Explanation
Command_Line_1	A253	First part of the command string
Command_Line_2	A253	Second part of the command string
Command_Line_3	A253	Third part of the command string
Command_Line_4	A253	Fourth part of the command string
Command_Line_5	A253	Fifth part of the command string
First-Call	L	Must be true for the first call of an UNLOAD command
P-NUMBERS		Returns the number of processed objects
Num_COPY	N5	Copycode
Num_CP	N5	Command processors
Num_DDM	N5	DDMs
Num_ERR	N5	Error messages
Num_FDT	N5	FDTs
Num_GLOB	N5	GDAs
Num_HELP	N5	Help routines
Num_LOC	N5	LDAs
Num_MAP	N5	Maps
Num_NAT_OBJ	N5	Total number of Natural objects
Num_PAR	N5	PDAs
Num_PRED_RLS	N5	Rules
Num_PROG	N5	Programs
Num_SUBP	N5	Subprograms
Num_SUBR	N5	Subroutines
Num_TEXT	N5	Text
Num_TOT	N5	Total number of objects

Parameter	Format/Length	Explanation
Num_CLASS	N5	Classes
Num_Dialog	N5	Dialogs
Return_Message	A253	Returned message text; always filled.
Return_Code	I4	<p>Return code. The following codes are possible:</p> <p>0 No warning or error occurred.</p> <p>-4870 Error in command; see Cmd_Return_Code. any other negative values are Natural errors.</p> <p>Any other values are error messages issued by SYSTRANS.</p>
Cmd_Return_Code	I4	<p>Return code from command analysis:</p> <p>0 No warning or error occurred</p> <p>1 No SYSTRANS command</p> <p>2 Command too long</p> <p>3 Invalid function</p> <p>4 Invalid object</p> <p>5 Invalid entry in command</p> <p>6 Value too long</p> <p>7 Invalid value (see message)</p> <p>8 UNLOAD was not followed by END</p> <p>9 Options in secondary UNLOAD</p> <p>10 Function rejected by user exit</p>
EXTENSIONS	(A1/1:V)	For future use.

An example of how to call TRANSIF can be found in the program DOCIF in library SYSTRANS.

Available Keywords and Functions

The following tables include all available keywords and values for the execution of SYSTRANS functions in direct command mode, that is, by using the direct command TRANSCMD or by using the Natural subprogram TRANSIF.

Depending on the specified function (see Functions below), keywords are either mandatory or optional. In the latter case, they are included in brackets []. Keywords separated by a "|" character represent alternatives.

An underlined portion of a keyword represents an alternative for the complete keyword. For example, you can specify EX instead of EXECUTE, or U instead of UNLOAD.

All default values correspond to those described in the corresponding sections of the SYSTRANS interactive usage; see also the sections Unload Function and Load, Scan and Restart Load Functions. They determine the value used if no keyword and value are specified. Keywords that do not have a default value are mandatory.

Functions

The following functions can be specified:

[<u>EX</u> ECUTE]	keyword (can be omitted)
<u>U</u> NLOAD <u>L</u> OAD <u>S</u> CAN <u>R</u> ESTART	mandatory values
[END]	keyword

The function block must be specified as the first parameter block of the command string.

If the UNLOAD function is specified, also the keyword END must be specified as last function of the block in order to write the end record to the work file (except if only one UNLOAD command is specified and the ENDREC option is set to Y; see also Options below). With any other function, END can be omitted in online mode.

Objects

The following objects can be specified:

[OBJECT]	keyword (can be omitted)
<u>N</u> AT-OBJECT <u>M</u> AP <u>D</u> DM <u>F</u> DT <u>E</u> RROR-MSG <u>A</u> LL <u>C</u> OMMAND-PROCESSOR	values

The object block must be specified immediately after the function block, except for the function RESTART. Valid values depend on the specified function.

All values (except ALL) are valid for the functions: UNLOAD, LOAD and SCAN. The value ALL is valid for the functions LOAD and SCAN only.

Parameters

The following parameters can be specified:

[WHERE]	Can be used to introduce the parameters block to make the command string more readable.
FROM FROM-LIBRARY	Source library when unloading Natural objects, maps, DDMs or user messages.
[NAME]	Object name when unloading, loading or scanning Natural objects, maps or DDMs.
[TYPE]	Object type when unloading, loading or scanning Natural objects; message type (<u>S</u> YSTEM <u>U</u> SER) when unloading, loading or scanning error messages.
[TO-LIBRARY]	Target library when unloading Natural objects, maps, DDMs or user messages.
[INCORPORATE]	Specifies whether to incorporate rules when unloading maps.
[UNLOAD-RULES]	Specifies whether to unload rules when unloading maps.
[RULE-LOAD]	Specifies whether to load rules when unloading maps.
FDBID	Source database ID when unloading FDTs.
FFNR	Source file number when unloading FDTs.
[TDBID]	Target database ID when unloading, loading or scanning FDTs.
[TFNR]	Target file number when unloading, loading or scanning FDTs.
[PASSWORD]	Password to be specified when unloading FDTs.
[CIPHER]	Cipher code to be specified when unloading FDTs.
[FNUMBER]	Start value when unloading, loading or scanning error messages.
[TNUMBER]	End value when unloading, loading or scanning error messages.
[LANGUAGE]	Message language when unloading, loading or scanning error messages.
[LIBRARY <u>L</u> IB]	Library when loading or scanning Natural objects, maps, DDMs or user messages.
[NEW-LIBRARY NEW-LIB]	New library when loading Natural objects, maps, DDMs or user messages.

The parameters block is mandatory with the UNLOAD function (except if the option WORK-FILE-INPUT is set to Y in the options block; see Options below). It can be omitted with the LOAD and SCAN functions, and it is not used with the RESTART and END functions. Valid keywords and possible values depend on the specified function.

Options

The following options can be specified:

[WITH]
[CONVERSION]
[USER-TABLE]
[SUBSTITUTE-REFERENCE]
[REPORT]
[LINE-NUMBERS]
[WORK-FILE-INPUT]
[WF-NAME]
[SPECIAL-CONVERSION]
[TRANSLATE-TO-UPPER]
[SAVE-RESTART-INFO]
[USE-NTC-WORK-FILE]
[REPLACE]
[ENDREC]

SPECIAL-CONVERSION only applies to the Load function and is the equivalent of User-Defined Conversion Table = F (Force) in CUI environments, and Use ASCII-ASCII Conversion Table in GUI environments.

The options block is optional and can be introduced by the keyword WITH to make the command string more readable. Valid keywords and possible values depend on the specified function and correspond to the SYSTRANS General Options for the individual functions as described in the sections Unload Function and Load, Scan and Restart Load Functions.

When using the UNLOAD function, options are evaluated during the first call only and must not be changed during subsequent calls.

If WORK-FILE-INPUT is set to Y when unloading, only one call to unload objects will be issued, and the "*E" end record will be written to the work file after all UNLOAD commands are executed from Work File 2. Additional UNLOAD commands delete the contents of Work File 1 and overwrite the data.

If REPORT is set to Y, the report to be displayed will be written to Work File 4, which is overwritten with each call of TRANSIF or TRANSCMD. Work file 4 should therefore not be used for other purposes by your application.

If you want to create a report for subsequent calls, you have to call TRANSIF from within a Natural program that opens Work File 4, for example, with: IF FALSE WRITE WORK 4 RECORD.

SYSTRANS Profile

You can define a profile for your SYSTRANS utility, both generally and user-specific. To this purpose, Natural is delivered with the text object PROFILE. To activate this profile, you have to resave PROFILE under the name TRANPROF in library SYSTRANS.

If you specify the UNLOAD or LOAD or SCAN functions, SYSTRANS searches for (and invokes) object TRANPROF in library SYSTRANS. In TRANPROF, you can enter general or user-specific profiles with corresponding function parameter defaults. These function parameter defaults are then displayed when you enter the function menu.

This section covers the following topics:

- Listing of Text Object PROFILE
- Parameters in the Text Object PROFILE

Listing of Text Object PROFILE

```
*****
* Application: SYSTRANS (TRANSFER)
* Object:      PROFILE
*****
* Function:    Source of the SYSTRANS Profile TRANPROF
*
*   To activate save as 'TRANPROF' in Library SYSTRANS.
*
*   Used to set defaults for Unload or Load/Scan General Options.
*
*   For possible values see the '[General-Start]' to '[General-End]'
*   block, it defines the options for all users.
*
*   In the '[User-Start uid]' to '[User-End uid]' blocks (where 'uid'
*   is the user ID as contained in the Natural system variable '*USER')
*   it is possible to define options for single users.
*   See example in the '[User-Start UID-EXAM]' to '[User-End UID-EXAM]'
*   block.
*   Notes:
*     - Empty lines or lines starting with '*' or '/*' are ignored.
*     - any text after '/*' is ignored.
*     - The line length must not exceed 250 bytes,
*       in mainframe environment only 90 bytes are allowed
*****
[General-Start]
[Unload-Start]
  Conversion N
  User-Conversion N
  Substitute-Line-References N
  Report Y
  Include-Line-Numbers N
  Work-File-Input N
  Use-Selection-List N
  /* Work-File-Name TRANWRK1    /* remove '/*' to use this option
  Use-NTC-Work-File N
  /* Unload-Library SYSTEM      /* remove '/*' to use this option
[Unload-End]
```



```
[Load-Start]
  User-Defined-Conversion-Table N
  Special-Table N
  Report Y
  Translate-to-Upper-Case N
  Save-Restart-Information N
  Modify-Library-Names N
  /* Work-File-Name TRANWRK1    /* remove '/*' to use this option
  Use-NTC-Work-File N
  Load-Library *
  Replace N
[Load-End]
[General-End]

[User-Start UID-EXAM]
  [Unload-Start]
    Report Y
  [Unload-End]
  [Load-Start]
    Translate-to-Upper-Case N
  [Load-End]
[User-End UID-EXAM]
```

Parameters in the Text Object PROFILE

The general SYSTRANS profile can be set between [General-Start] and [General-End]. Settings entered here apply to all users. User-specific settings can be entered between [User-Start UID] and [User-End UID], where UID is the Natural user ID (*USER).

Generally and for a user, you can set parameters for the UNLOAD function. You enter these parameters between [Unload-Start] and [Unload-End]. For the LOAD and SCAN functions, you enter the parameters between [Load-Start] and [Load-End]. The parameters for the UNLOAD and LOAD/SCAN functions are described in detail below.

Unload Parameters

Parameters for the general UNLOAD options:

Conversion	possible values: Y (Yes), N (No)
User-Conversion	possible values: Y,N
Substitute-Line-References	possible values: Y,N
Report	possible values: Y,N
Include-Line-Numbers	possible values: Y,N
Work-File-Input	possible values: Y,N
Use-Selection-List	possible values: Y,N
Work-File-Name	possible values: Name of Work File 1 (only on OpenVMS, UNIX and Windows)
Use-NTC-Work-File	possible values: Y,N

Other UNLOAD parameters:

Unload-Library	possible values: Default library name
----------------	---------------------------------------

Load Parameters

Parameters for the general LOAD options:

User-Defined-Conversion-Table	possible values: Y (Yes), N (No)
Special-Table	possible values: Y,N (corresponds to the character version setting "User-Defined Conversion Table=F")
Report	possible values: Y,N
Translate-to-Upper-Case	possible values: Y,N
Save-Restart-Information	possible values: Y,N
Modify-Library-Names	possible values: Y,N
Work-File-Name	possible values: Name of Work File 1 (only on OpenVMS, UNIX and Windows)
Use-NTC-Work-File	possible values: Y,N

Other LOAD parameters:

Load-Library	possible values: Default library name
Replace	possible values: Y,N

Downloading Objects without SYSTRANS

Not applicable to mainframes.

A downloaded object must be made known to Natural. This is done with either of the following methods:

- the Import function of the SYSMAIN utility
(see the Natural User's Guide for OpenVMS and UNIX).
- Using the utility FTOUCH
- Using NFS to Store Natural Libraries

Using the Utility FTOUCH

 To execute the FTOUCH utility

1. Go to an operating system command prompt.
2. Ensure that the transferred file is in the desired FNAT or FUSER directory (as specified in your global configuration file) and has the correct extension.
3. Enter the command **ftouch**.

ftouch has the following syntax, the parts of which are described below:

```
ftouch [fnat=dbid,fnr] [fuser=dbid,fnr][bp=bp-name]  
[parm=parm-file] [lib=library-name]  
[-v] [mode] [kind] files
```

The FTOUCH utility can also be used for migration purposes, in which case it has to be invoked as follows:

```
ftouch [fnat=dbid,fnr] [fuser=dbid,fnr]  
[parm=parm-file] [lib=library-name] convert
```

Note:

Terms enclosed in brackets ([]) are optional; bold letters are actual values that must be entered as shown. The following options are provided:

Option	Explanation
fnat = <i>dbid,fnr</i>	Specifies the database ID and file number of the FNAT system file to be used; default is the value specified in NATPARM.
fuser = <i>dbid,fnr</i>	Specifies the database ID and file number of the FUSER system file to be used; default is the value specified in NATPARM.
bp = <i>bp-name</i>	Specifies the buffer pool to be used. You can omit the <i>bp-name</i> if you want to use the Natural default buffer pool NATBP; otherwise, you have to specify the appropriate <i>bp-name</i> . Note: If the Natural default buffer pool is not active or if the specified buffer pool does not exist, an appropriate error message is displayed.
parm = <i>parm-name</i>	Specifies the name of the parameter file to be used if you want to use a parameter file other than the default parameter file NATPARM.
lib = <i>library-name</i>	Specifies the library to be used. You can omit the <i>library-name</i> if you are already in the appropriate subdirectory; otherwise you have to specify the appropriate <i>library-name</i> .

Option	Explanation
userep=ON/OFF	Windows only. Specifies whether you want to use the repository or not.
-v	Displays statistics on disk I/Os during processing.
<i>mode</i>	Specifies the programming mode; sm specifies that a program is in structured mode; the default is reporting mode.
<i>kind</i>	Specifies the subdirectory for input; it can be one of the following: <ul style="list-style-type: none"> -s for source programs (default), -g for generated programs, -b for both source and generated programs.
<i>files</i>	Specifies the files to be processed; you can specify <i>filename.ext</i> for individual files or: <ul style="list-style-type: none"> -a to add new files; all files in the directory which are currently found in FILEDIR.SAG are added (already existing files are not touched). -d to build a new FILEDIR.SAG directory. <p>Attention: Be careful when using this option, since the old FILEDIR.SAG is deleted and rebuilt from scratch.</p>
-f	Forces an update of the specified object's timestamp in FILEDIR.SAG. This option can only be specified if an individual file has been specified with the <i>files</i> option (see above).
convert	Indicates that the specified library and system files are to be migrated from Natural Version 2.1 to Natural Version 2.2; the old FILEDIR.SAG file is renamed to FILEDIR.212 and a new FILEDIR.SAG file is created.
sync	Indicates that the specified library and system files are to be synchronized between Natural and the repository (Windows only); this function must be executed each time FILEDIR.SAG is modified by FTOUCH. <p>Attention: When specifying sync, ensure that either "userep=ON" is set or the Natural profile parameter USEREP is set to ON.</p>

Example 1:

Change to the following directory: *fuser-directory/TESTLIB/SRC*

Enter the following command: **ftouch sm TESTFILE.NSP**

As a result, the program TESTFILE in library TESTLIB is available in structured mode to Natural.

Example 2:

Change to the following directory: *fuser-directory/MYLIB*

Enter the following command: **ftouch fnat=21,21 fuser=22,22 -b**

As a result, all files in the directories MYLIB/SRC and MYLIB/GP are available in reporting mode (default) to Natural.

Using NFS to Store Natural Libraries

When you use NFS to store Natural libraries, you can run into problems when the directories in which the Natural libraries are stored are mounted via NFS from a file server in your network.

The reason for this is the need to lock the FILEDIR.SAG file stored in each library during update operations of Natural objects.

If your NFS locking is incompatible or not properly set up between the involved platforms, Natural can hang in an uninterruptible state while waiting for NFS locking requests to be processed. These requests are generally logged on the consoles of the involved systems or in some other system-dependent log file.

The work-around to solve this problem is to store Natural libraries only on local disks if problems with a hanging and uninterruptible nucleus occur.